


[Subscribe \(Full Service\)](#) [Register \(Limited Service, Free\)](#) [Login](#)

 Search: ☐ The ACM Digital Library ☒ The Guide


**THE GUIDE TO COMPUTING LITERATURE**

[Feedback](#) [Report a problem](#) [Satisfaction survey](#)

 Terms used [transfer point jit](#)

Found 42 of 902,478

Sort results by


[Save results to a Binder](#)
[Try an Advanced Search](#)

Display results


[Search Tips](#)
[Try this search in The Digital Library](#)
☐ Open results in a new window

Results 1 - 20 of 42

 Result page: [1](#) [2](#) [3](#) [next](#)

 Relevance scale ☐ ☐ ☐ ☐ ☐
**1 [A region-based compilation technique for a Java just-in-time compiler](#)**


Toshio Suganuma, Toshiaki Yasue, Toshio Nakatani

 May 2003 **ACM SIGPLAN Notices , Proceedings of the ACM SIGPLAN 2003 conference on Programming language design and implementation PLDI '03**, Volume 38 Issue 5

Publisher: ACM Press

Full text available: pdf(158.62 KB)

 Additional Information: [full citation](#), [abstract](#), [references](#), [citations](#), [index terms](#)

Method inlining and data flow analysis are two major optimization components for effective program transformations, however they often suffer from the existence of rarely or never executed code contained in the target method. One major problem lies in the assumption that the compilation unit is partitioned at method boundaries. This paper describes the design and implementation of a region-based compilation technique in our dynamic compilation system, in which the compiled regions are selected a ...

**Keywords:** dynamic compilers, on-stack replacement, partial inlining, region-based compilation

**2 [Partial method compilation using dynamic profile information](#)**


John Whaley

 October 2001 **ACM SIGPLAN Notices , Proceedings of the 16th ACM SIGPLAN conference on Object oriented programming, systems, languages, and applications OOPSLA '01**, Volume 36 Issue 11

Publisher: ACM Press

Full text available: pdf(1.73 MB)

 Additional Information: [full citation](#), [abstract](#), [references](#), [citations](#), [index terms](#)

The traditional tradeoff when performing dynamic compilation is that of fast compilation time versus fast code performance. Most dynamic compilation systems for Java perform selective compilation and/or optimization at a method granularity. This is the not the optimal granularity level. However, compiling at a sub-method granularity is thought to be too complicated to be practical. This paper describes a straightforward technique for performing compilation and optimizations at a finer, sub-metho ...

**3 [Papers: Open signaling for ATM, internet and mobile networks \(OPENSIG'98\)](#)**


Andrew T. Campbell, Irene Katzela, Kazuho Miki, John Vicente

 January 1999 **ACM SIGCOMM Computer Communication Review**, Volume 29 Issue 1

Publisher: ACM Press